CNN for Device Identification

Kahraman Kostas

The CNN method is used in network security or classification of network data, while the network packet information is converted to pseudo-images to present to the algorithm. Some of the paths followed in these conversion processes are mentioned below.

Lim and his colleagues [1] converted the network data into pseudo-images for use on CNN. For this process, firstly, the payload portion of the network packet was converted to binary numbers to obtain 4-bit groups (two-part for every byte). These nibbles were converted to decimal numbers, each of which acted as pixels of a picture. Pseudo-pictures of 36,64,256,1024 pixels were produced according to their size in payloads (See Figure 1).

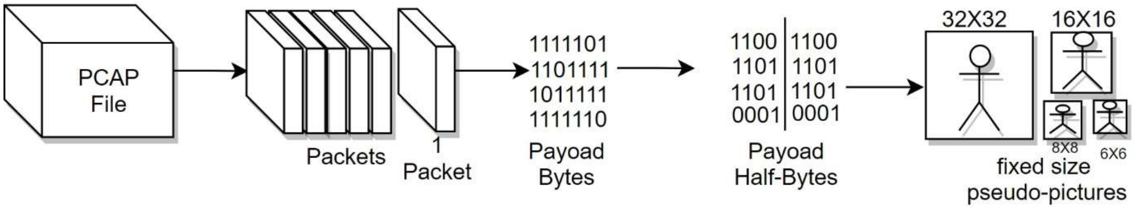


Figure 1 Representation of the operation of the algorithm simply.

For this operation, the length of the series of numbers to create the image is rounded to the nearest image size (36,64,256 or 1024). For this, the interrupt or 0 padding method is applied. For example, the algorithm adds 26 zeros to a series of 10 numbers to reach the closest length option (36). On the other hand, in a series with a length of 500 numbers, it is ignored by cutting after 256. Some of the pseudo pictures obtained from this process (pictures from the first package for 10 different devices) are shared below (See Table 1).

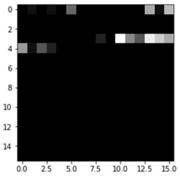
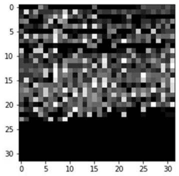
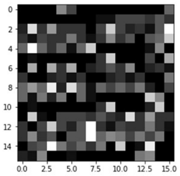
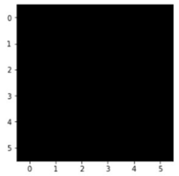
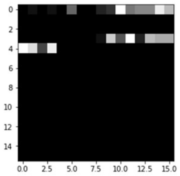
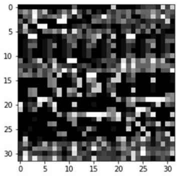
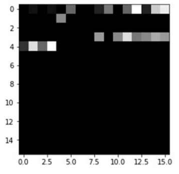
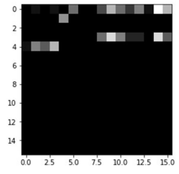
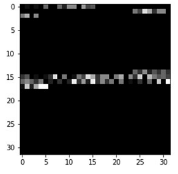
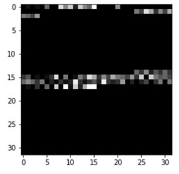
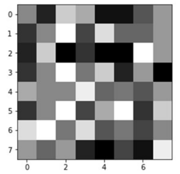
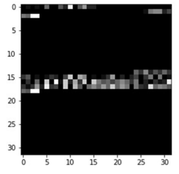
					
Aria	D-LinkCam	D-LinkDayCam	D-LinkDoorSensor	D-LinkHomeHub	D-LinkSensor
					
D-LinkSwitch	D-LinkWaterSensor	EdimaxCam	EdimaxPlug1101W	HomeMaticPlug	HueBridge

Table 1 Pictures from the first package for 10 different devices.

Lotfollahi et al[2] use pcap files for CNN and SAE. These operations can be listed as follows. The Ethernet header is deleted. Standard packets such as DNS or tree-way handshake packets are deleted because they do not carry payloads and are not effective in determining the class of data. Adding 0 to the end of UDP packets and IP payloads less than 1480 Bytes ensures that all data arrays are the same size. Then normalization is performed by dividing each byte to 255. The result is a pseudo-image of 1480 pixels (37x40).

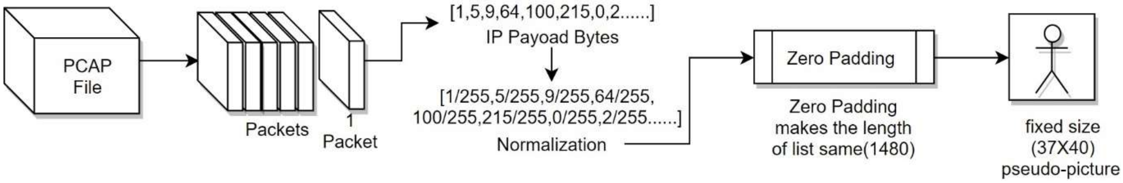


Figure 2 Representation of the operation of the algorithm simply.

Some of the pseudo pictures obtained from this process (pictures from the first package for 10 different devices) are shared below (See Table 2).

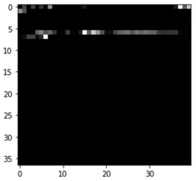
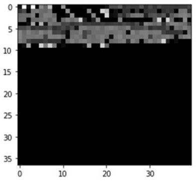
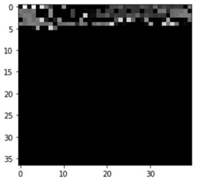
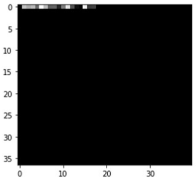
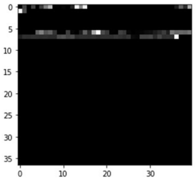
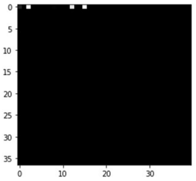
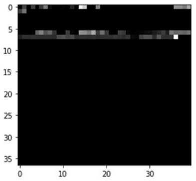
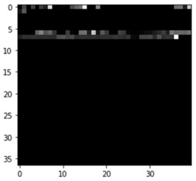
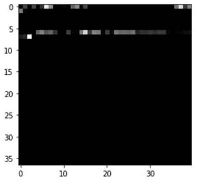
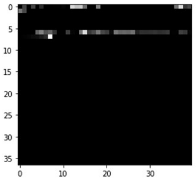
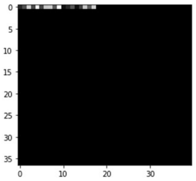
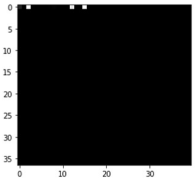
					
Aria	**D-LinkCam**	**D-LinkDayCam**	**D-LinkDoorSensor**	**D-LinkHomeHub**	**D-LinkSensor**
					
D-LinkSwitch	**D-LinkWaterSensor**	**EdimaxCam**	**EdimaxPlug1101W**	**HomeMaticPlug**	**HueBridge**

Table 2 Pictures from the first package for 10 different devices.

In their study, **Wang et al [3]** convert 4 different data groups (Session + All, Session + L7, Flow + All, Flow + L7) to pseudo-pictures of 784 (28 * 28) pixels to be used in CNN. In this process, Session and flow refer to the 5-tuples such as source IP, source port, destination IP, destination port and transport-level protocol. However, while the session is bi-directional, flow is one-way. L7 stands for application layer and ALL stands for all protocol layers.

We have performed only one (Session + L7) of these options and shared the methods and results below. The first 13 pixels of the generated image symbolize the session (Source IP:4 Byte, Destination IP:4 Byte, Source Port:2 Byte, Destination Port:2 Byte, UDP/TCP:1 Byte) and the remaining 771 pixels symbolize the payload.

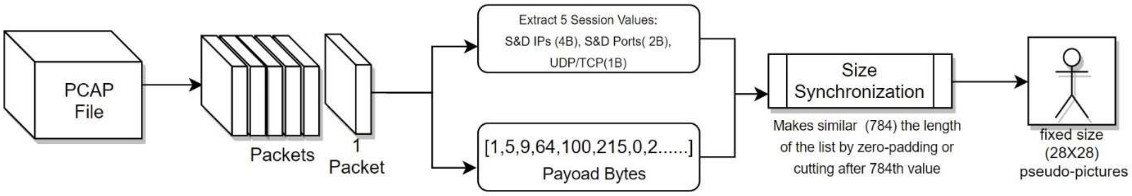

PCAP File
Packets
1 Packet
Extract 5 Session Values:
S&D IPs (4B), S&D Ports(2B),
UDP/TCP(1B)
[1,5,9,64,100,215,0,2......]
Payload Bytes
Size Synchronization
Makes similar (784) the length of the list by zero-padding or cutting after 784th value
fixed size (28X28) pseudo-pictures

Figure 3 Representation of the operation of the algorithm simply.

Some of the pseudo pictures obtained from this process (pictures from the first package for 10 different devices) are shared below (See Table 3).

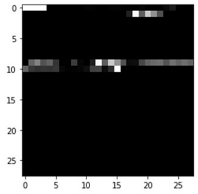
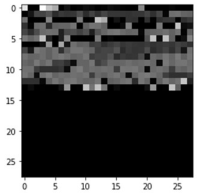
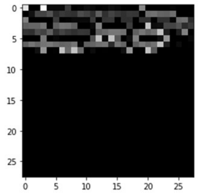
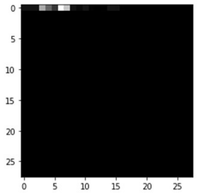
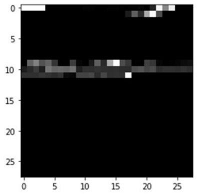
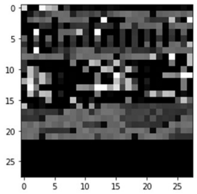
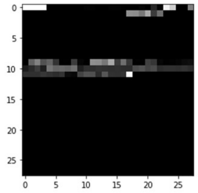
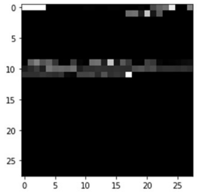
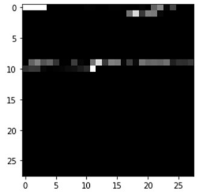
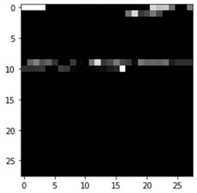
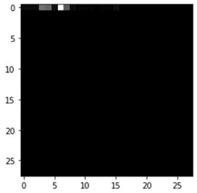
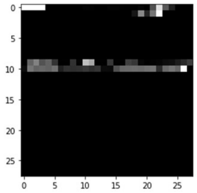
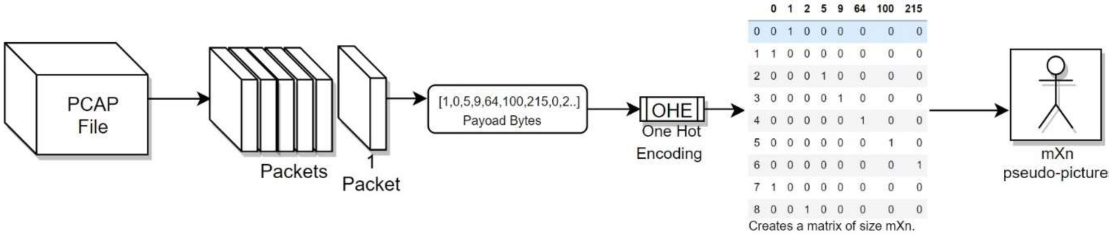
					
Aria	D-LinkCam	D-LinkDayCam	D-LinkDoorSensor	D-LinkHomeHub	D-LinkSensor
					
D-LinkSwitch	D-LinkWaterSensor	EdimaxCam	EdimaxPlug1101W	HomeMaticPlug	HueBridge

Table 3 Pictures from the first package for 10 different devices.

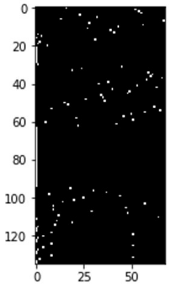
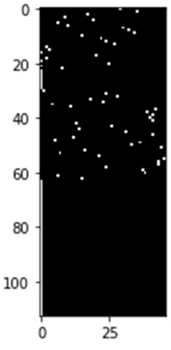
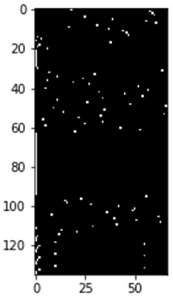
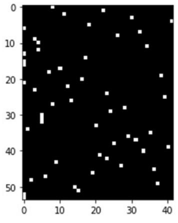
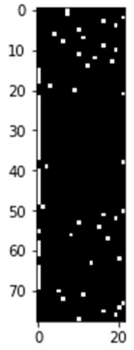
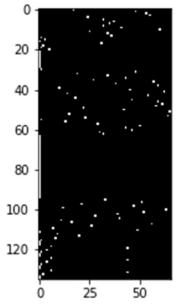
In another study by **Wang et al[4],** Each network packet is converted to a two-dimensional matrix for use in CNN. For this process, the headers are removed from the network packet. The One Hot Encoding method is applied to the remaining payload. As a result of this method, a binary matrix of *mxn* size is created from the array of length m (payload). Here, n represents the values of the bytes in the packet. In this method, unlike other methods, the size of the images is not predicted.



	0	1	2	5	9	64	100	215
0	0	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
2	0	0	0	1	0	0	0	0
3	0	0	0	0	1	0	0	0
4	0	0	0	0	0	1	0	0
5	0	0	0	0	0	0	1	0
6	0	0	0	0	0	0	0	1
7	1	0	0	0	0	0	0	0
8	0	0	1	0	0	0	0	0

Figure 4 Representation of the operation of the algorithm simply.

Some of the pseudo pictures obtained from this process (pictures from the first package for 10 different devices) are shared below (See Table 4).

					
Aria	D-LinkCam	D-LinkDayCam	D-LinkDoorSensor	D-LinkHomeHub	D-LinkSensor

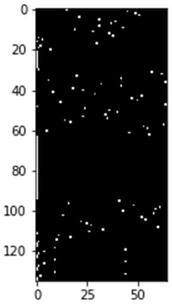
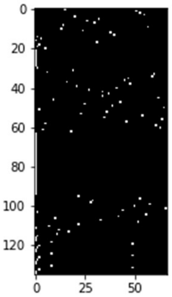
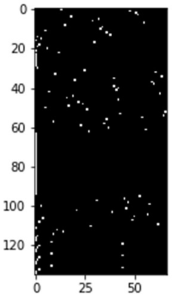
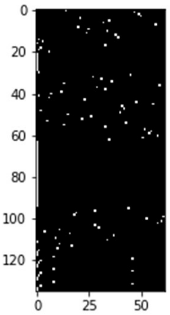
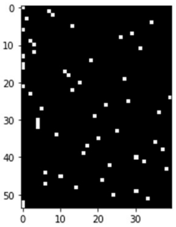
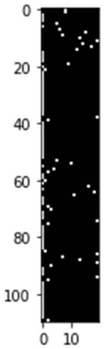
					
D-LinkSwitch	D-LinkWaterSensor	EdimaxCam	EdimaxPlug1101W	HomeMaticPlug	HueBridge

Table 4 Pictures from the first package for 10 different devices.

[1] H.-K. Lim, J.-B. Kim, J.-S. Heo, K. Kim, Y.-G. Hong, and Y.-H. Han, "Packet-based Network Traffic Classification Using Deep Learning," in 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2019, pp. 046-051: IEEE.

[2] M. Lotfollahi, R. Shirali, M.J. Siavoshani, and M. Saberian, “Deep Packet: A Novel Approach For Encrypted Traffic Classification Using Deep Learning,” arXiv preprint arXiv:1709.02656 (2017).

[3] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, “End-to-end encrypted traffic classification with one-dimensional convolution neural networks,” In Intelligence and Security Informatics (ISI), IEEE International Conference on. IEEE, Jul. 2017, pp. 43-48.

[4] W. Wang, et al, “HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection,” IEEE Access, vol. 6, 2018, pp. 1792-1806.